Applic. No. 10/620,108
Amdt. dated July 10, 2007
Reply to Office action of April 10, 2007

Claim Amendments

This listing of the claims will replace all prior versions,
and listings, of claims in the application:

Claim 1 (currently amended): A security module for use with a
terminal, comprising

a data interface adapted to be coupled to a terminal, for
receiving ~~at least~~ one of part of an algorithm code ~~or of~~ and
the complete algorithm code from the terminal, with the
algorithm code concerning a processing of secrets[[,]] ;

~~an energy~~ a power interface for receiving supply ~~energy~~ power
from the terminal;

a volatile memory for storing the one of the part of the
algorithm code ~~or~~ and the complete algorithm code received via
the data interface, said volatile memory being coupled to the
~~energy~~ power interface in order to have ~~energy~~ power supplied
thereto such that ~~the same~~ said volatile memory will be
cleared upon an interruption of the receipt of the supply
~~energy~~ power from the terminal; and

Applic. No. 10/620,108
Amdt. dated July 10, 2007
Reply to Office action of April 10, 2007

a processor for performing the algorithm code in order to

obtain an algorithm code result that can be delivered to the

terminal.

Claim 2 (currently amended). A security module according to

claim 1, wherein the data interface is adapted to receive only

the part of the algorithm code, the security module further

comprising:

a non-volatile memory in which ~~the non-received~~ a remainder

of the algorithm code is stored along with the received part

of the algorithm code for forming the complete algorithm code.

Claim 3 (original): A security module according to claim 1,

further comprising:

a means for performing an authentication between the terminal

and the security module.

Claim 4 (currently amended): A security module according to

~~any of~~ claim 1, wherein the data interface is arranged to

receive from the terminal ~~said~~ the one of the part of the

algorithm code ~~or~~ and the complete algorithm code in encrypted

form ~~and/or~~ and with a certificate, with the security module

further comprising:

Applic. No. 10/620,108
Amdt. dated July 10, 2007
Reply to Office action of April 10, 2007

a means for decrypting ~~said~~ <u>the one of the</u> part of the

encrypted algorithm code ~~or~~ <u>and</u> the encrypted complete

algorithm code; and

a means for examining the certificate and for preventing

performing of the algorithm code ~~if~~ <u>depending on examination</u>

<u>of</u> said certificate ~~lacks genuineness~~.

Claim 5 (currently amended): A security module according to

~~any of~~ claim 1, <u>wherein said data interface is adapted to</u>

<u>receive only the part of the algorithm code, the security</u>

<u>module</u> further comprising:

a memory managing unit for controlling memory accesses of the

processor, with the ~~transferred~~ part of the algorithm code

containing addresses of the algorithm code.

Claim 6 (currently amended): A security module according to

~~any of~~ claim 1, further comprising:

a means for monitoring a predetermined security condition and

for clearing the volatile memory if said predetermined

security condition is fulfilled, with said security condition

being selected from a plurality of conditions comprising an

interruption, an irregularity and a fluctuation of the supply

voltage and of a system clock as well as of additional

operating parameters.

Claim 7 (currently amended): A security module according to
~~any of~~ claim 1, wherein the algorithm code comprises a program
code selected for carrying out a task selected from a group
~~comprising~~ consisting of a symmetric cryptographic algorithm,
an asymmetric cryptographic algorithm, an RSA algorithm, a
cryptographic process according to the DES standard, an
elliptic curve process and an access function for accessing a
digital value stored on the security module as well as an
access function for changing [[a]] the digital value stored on
the security module.

Claim 8 (currently amended): A security module according to
~~any of~~ claim 1, wherein said data interface is adapted to
receive only the part of the algorithm code, the part ~~received~~
of the algorithm code ~~comprises~~ comprising a start address of
the algorithm code, memory addresses of computing components
necessary for performing the algorithm code, or jump addresses
of the algorithm code.

Claim 9 (currently amended): A security module according to
~~any of~~ claim 1, wherein the data interface is adapted to

Applic. No. 10/620,108
Amdt. dated July 10, 2007
Reply to Office action of April 10, 2007

<u>receive the one of the part of the algorithm code and the</u>

<u>complete algorithm code several times in different versions,</u>

<u>with</u> the volatile memory ~~is~~ <u>being</u> arranged for storing a newly

received~~, altered part~~ <u>version of one of the part of the</u> of

the algorithm code <u>and the complete algorithm code</u> ~~over~~ <u>such</u>

<u>that</u> the ~~stored~~ <u>previously received version of one of the</u> part

of the algorithm ~~code or the stored~~ <u>and the</u> complete algorithm

code <u>is overwritten</u>.

Claim 10 (currently amended): A security module according to

~~any of~~ claim 1, wherein said security module is designed as a

chip card.

Claim 11 (currently amended): A process for computing an

algorithm code result using a security module, comprising the

steps of:

receiving ~~at least~~ <u>one of</u> part of an algorithm code ~~or~~ <u>and</u> the

complete algorithm code by means of an ~~energy~~ interface <u>to a</u>

<u>terminal</u>, with the algorithm code concerning a processing of

secrets;

~~volatile storing said~~ <u>storing the one of the</u> part of the

algorithm code ~~or said~~ <u>and the</u> complete algorithm code in a

volatile memory of the security module, with the volatile

memory being coupled to the ~~energy~~ interface, to be supplied

with ~~energy~~ power, such that the ~~same~~ <u>volatile memory</u> will be

cleared upon an interruption of the receipt of the supply

~~energy~~ power from the terminal[[:]] <u>;</u>

performing said algorithm code on the security module in order

to obtain an algorithm code result; <u>and</u>

delivering said algorithm code result to the terminal~~, and~~

~~clearing said volatile memory upon an interruption of the~~

~~receipt of the supply energy from the terminal~~.

Claim 12 (currently amended): A process according to claim

11, ~~wherein said step of clearing comprises~~ <u>further comprising</u>

removing the security module from the terminal <u>thereby causing</u>

<u>an interruption of the receipt of the supply power to the</u>

<u>volatile memory from the terminal and clearing said volatile</u>

<u>memory upon interruption of the receipt of supply power from</u>

<u>the terminal</u>.

Claim 13 (currently amended): A terminal for use with a

security module, comprising:

Applic. No. 10/620,108
Amdt. dated July 10, 2007
Reply to Office action of April 10, 2007

a data interface adapted to be coupled to the security module,

for transmitting at least part of an algorithm code or the

complete algorithm code from the terminal to a volatile memory

of the security module and for receiving ~~the~~ an algorithm

code result from the security module, with the algorithm code

concerning a processing of secrets; and

~~an energy~~ a power interface for delivering supply ~~energy~~ power

to the security module, with the volatile memory being

supplied by the supply ~~energy~~ power, such that the ~~same~~

volatile memory will be cleared upon an interruption of the

receipt of the supply ~~energy~~ power from the terminal,

with the terminal, for each communication operation between

terminal and security module ~~during one and the same~~

~~communication operation with the security module~~, being

~~designated~~ adapted to, during a single one of the

communication operations with the security module, send at

least the part of the algorithm code or the complete algorithm

code to the volatile memory of the security module; and,

~~subsequently, during the further communication process,~~ after

sending, receive the algorithm code result from the security

module.

Applic. No. 10/620,108
Amdt. dated July 10, 2007
Reply to Office action of April 10, 2007

Claim 14 (currently amended): A process for controlling within a plurality of communication operations, a security module using a terminal in order to obtain an algorithm code result from the security module, with the process comprising for each communication operation, performing the following steps during ~~one and the same~~ a single one of the communication ~~operation~~ operations with the security module:

delivering supply ~~energy~~ power from the terminal to the security module;

transmitting at least part of an algorithm code or the complete algorithm code from the terminal to a volatile memory of the security module, with the algorithm code concerning a processing of secrets, with the volatile memory being supplied by the supply ~~energy~~ power, such that the ~~same~~ volatile memory will be cleared upon an interruption of the receipt of the supply ~~energy~~ power from the terminal; and

receiving ~~the~~ an algorithm code result from the security module.

Claim 15 (currently amended): A process for communication between a security module and a terminal, comprising the steps of:

Applic. No. 10/620,108
Amdt. dated July 10, 2007
Reply to Office action of April 10, 2007

transferring ~~at least~~ one of part of an algorithm code ~~or~~ and

the complete algorithm code from the terminal to the security

module, with the algorithm code concerning a processing of

secrets;

~~volatile storing said~~ storing the one of the part of the

algorithm code ~~or~~ and said complete algorithm code in a

volatile memory of the security module, with the volatile

memory being supplied by ~~the~~ supply ~~energy~~ power, such that

the ~~same~~ volatile memory will be cleared upon interruption of

the receipt of the supply ~~energy~~ power from the terminal;

performing said algorithm code on the security module in order

to obtain an algorithm code result;

delivering said algorithm code result to the terminal; and

clearing said volatile memory upon an interruption of the

receipt of the supply ~~energy~~ power from the terminal.

Claim 16 (currently amended): A process according to claim

15, further comprising:

Applic. No. 10/620,108
Amdt. dated July 10, 2007
Reply to Office action of April 10, 2007

~~repeated~~ sequentially transferring ~~of~~ a plurality of different

versions of ~~said~~ the one of the part of the algorithm code ~~or~~

and said complete algorithm code; and

sequentially storing the ~~repeatedly transferred version~~ the

different versions of ~~said~~ the one of the part of the

algorithm code ~~or of~~ and the complete algorithm code ~~over the~~

~~stored part of the algorithm code or over the complete stored~~

~~algorithm code~~ such that a respective previous version of the

plurality of different versions of the one of the part of the

algorithm code and the complete algorithm code is overwritten.

Claim 17 (new): A security module for use with a terminal,

comprising:

a data interface adapted to be coupled to a terminal, for

receiving a first part of an algorithm code from the terminal,

with the algorithm code concerning a cryptographic processing

of data;

a power interface for receiving supply power from the

terminal;

a volatile memory for storing the first part of the algorithm

code received via the data interface, said volatile memory

Applic. No. 10/620,108
Amdt. dated July 10, 2007
Reply to Office action of April 10, 2007

being coupled to said power interface in order to have power

supplied thereto such that the volatile memory will be cleared

upon an interruption of the receipt of the supply power from

the terminal;

a non-volatile memory in which a second part of the algorithm

code which is a non-received remainder the algorithm code is

stored; and

a processor for performing the algorithm code in order to

obtain cryptographically processed data that can be delivered

to the terminal, wherein the first part of the algorithm code

includes memory addresses of computing components necessary

for performing the algorithm code, or jump addresses of the

algorithm code pointing to partial routines of the algorithm

code.